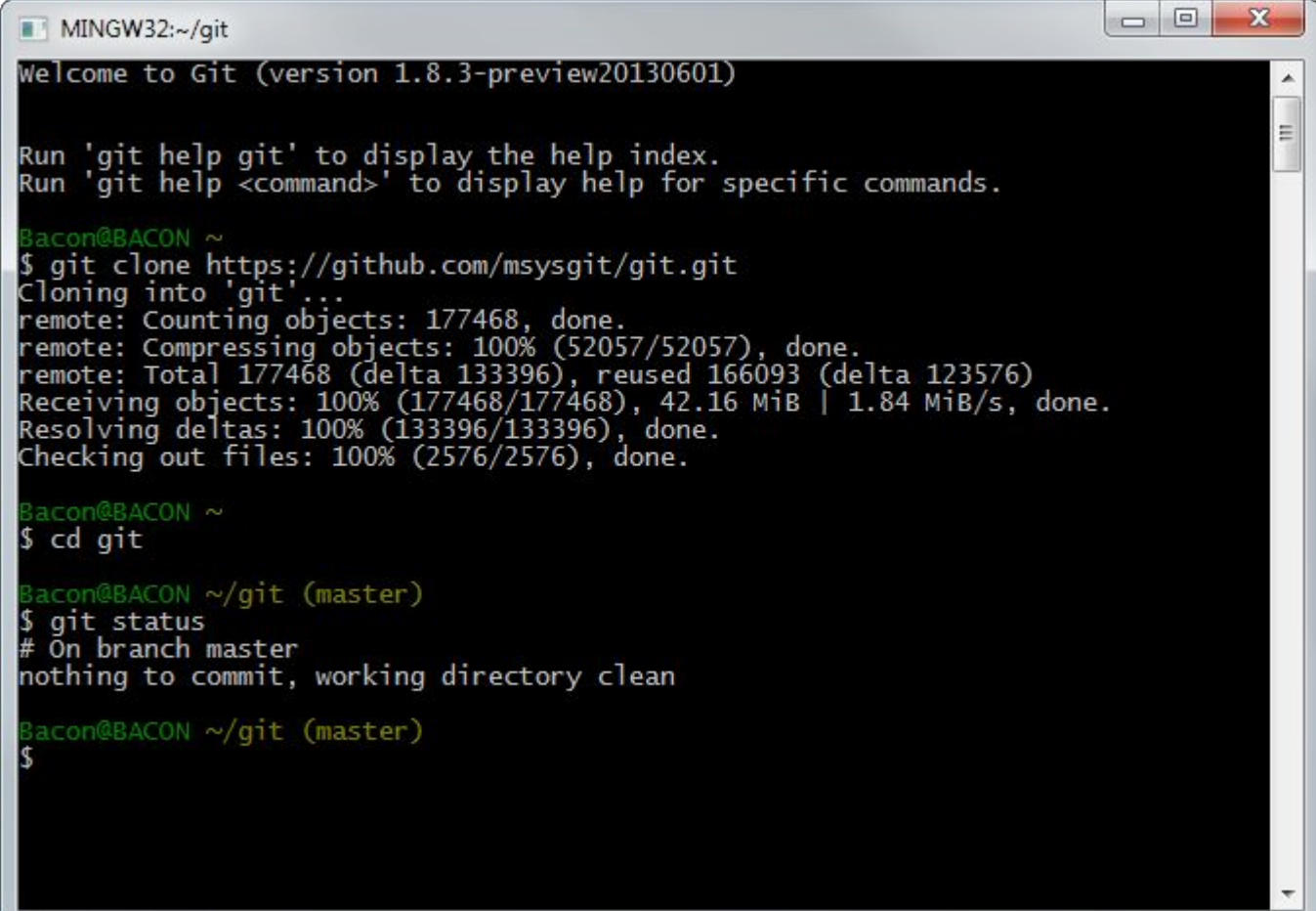


Uso del Terminal Git bash



```
MINGW32:~/git
Welcome to Git (version 1.8.3-preview20130601)

Run 'git help git' to display the help index.
Run 'git help <command>' to display help for specific commands.

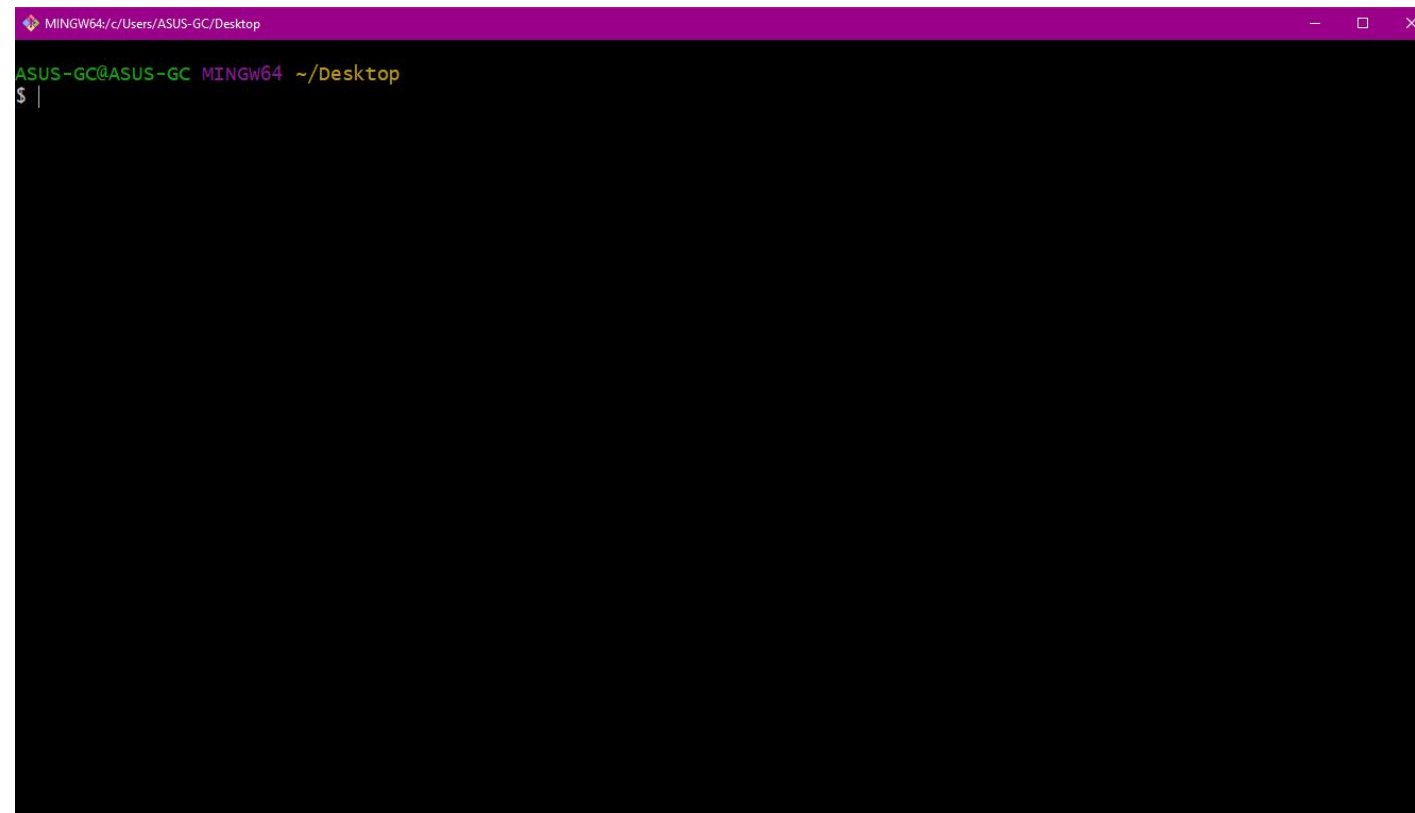
Bacon@BACON ~
$ git clone https://github.com/msysgit/git.git
Cloning into 'git'...
remote: Counting objects: 177468, done.
remote: Compressing objects: 100% (52057/52057), done.
remote: Total 177468 (delta 133396), reused 166093 (delta 123576)
Receiving objects: 100% (177468/177468), 42.16 MiB | 1.84 MiB/s, done.
Resolving deltas: 100% (133396/133396), done.
Checking out files: 100% (2576/2576), done.

Bacon@BACON ~
$ cd git

Bacon@BACON ~/git (master)
$ git status
# On branch master
nothing to commit, working directory clean

Bacon@BACON ~/git (master)
$
```

¿Que és Git bash?



Git Bash es una aplicación para entornos de Microsoft Windows que ofrece una capa de emulación para una experiencia de líneas de comandos de Git. Bash es el acrónimo en inglés de Bourne Again Shell. Una shell es una aplicación de terminal que se utiliza como interfaz con un sistema operativo mediante comandos escritos.

¿A tener en cuenta?

Estructura de archivos: Windows – Mac – Linux

- La ruta principal en Windows es `c:\` y en UNIX es solo `/`.
- Windows no hace diferencia entre mayúsculas y minúsculas, pero UNIX si.
- Git Bash usa la ruta `/c` para dirigirse al disco `c:/` en Windows. Por lo tanto, la ruta del usuario con la que vas a trabajar es `/c/Users/NombreUsuario`

Comandos básicos

1. **pwd:** nos muestra la carpeta actual en la que nos encontramos.
2. **mkdir:** nos permite crear carpetas, p. ej. mkdir NuevaCarpeta
3. **touch:** nos permite crear archivos nuevos, p.ej. touch NuevoArchivo.txt
4. **cat:** nos permite ver el contenido de un archivo, p.ej. cat NuevoArchivo.txt
5. **cd:** nos permite cambiarnos de carpeta, p.ej. cd NuevaCarpeta.
6. **cd ..:** nos permite regresar al directorio o carpeta anterior.
7. **cd o cd ~:** nos lleva a la ruta del usuario.
8. **cd /c:** nos vamos al disco C:/.
9. **cd -:** nos lleva directamente al ultimo directorio visitado.
10. **ls:** nos permite ver los archivos de la carpeta donde estamos actualmente.
11. **ls -l:** Ver todos los archivos como una lista en donde incluye el usuario, grupo, permisos sobre el archivo, tamaño, fecha y hora de creación.
12. **ls -lh:** Muestra la misma información que ls-l pero con unidades de tamaño, es decir, kb o mb.

13. **ls-R:** muestra el contenido de todos los subdirectorios de forma recursiva.
14. **ls -S:** Ordena los resultados por tamaño de archivo.
15. **rm:** Nos permite borrar un archivo o carpeta ej: rm NuevoArchivo.txt
16. **rmdir "nombre del directorio"/:** borrar un directorio: Solo funciona con directorios vacíos.
17. **rm -r 'nombre de la carpeta':** me permite eliminar la carpeta y los archivos dentro de ella de forma recursiva.
18. **cp "nombre del archivo que queremos copiar" "nombre del directorio a donde lo queremos copiar":** nos permite copiar un archivo.
19. **mv "el directorio de donde queremos mover/el nombre del archivo" "el directorio hacia donde lo queremos mover":** nos permite mover un archivo.
20. **clear:** nos permite limpiar la pantalla.
21. **history:** ver los últimos comandos que ejecutamos y un número especial con el que podemos volver a repetir el comando.

La única manera de perderle el miedo al Terminal es ...

¡USÁNDOLO!



<https://gitforwindows.org/>



¡DISFRUTA LO QUE HACES!

factoriaf5.org

rompemosloscodigos.org

